Federated Diffusion Model with Non-IID Data

Ruochen Jin

October 18, 2022

1 Literature Review

In this section, I'd like to introduce 5 popular diffusion models: DDPM, DDIM, improved DDPM, Latent Diffusion and Classifier-free Diffusion (in chronological order) [Weng, 2021].

Denoising Diffusion Probabilistic Model (DDPM) [Ho et al., 2020] improves the sample quality of GANs, flows, autoregressive models, and amplifies the impacts of generative models on the broader world. The authors simplified the loss by omitting variance $\sum_\theta$ (let $\sum_\theta(x_t, t) = \sigma^2 I$ be a constant)and introduced U-net with self-attention to estimate the loss between $q(x_t|x_0), t$. DDPMs can also achieve competitive log-likelihoods while maintaining high sample quality.

It's very slow to generate a sample from DDPM by following the Markov chain of a few thousand steps, there's a humorous gap in generation speed compared to GAN. Song et al. [Song et al., 2020] rewrite $q_\sigma(x_t - 1|x_t, x_0)$ to be parameterized by a desired standard deviation σ_t. This denoising diffusion implicit model (DDIM) has the same marginal noise distribution but the generative process is deterministic.

There's another approach is to run a strided sampling schedule [Nichol and Dhariwal, 2021], and it samples update every $\lceil T/S \rceil$ steps to reduce the process from T to S steps. In DDPM, neglected variance cannot be predicted, in contrast, the improved DDPM introduce a hybrid learning objective that combines the VLB with the simplified objective from DDPM. Another improvement is to use a cosine based variance schedule, which preserves more information during diffusion process.

Latent diffusion model (LDM; [Rombach et al., 2022]) runs the diffusion process in the latent space instead of pixel space, making training cost lower and inference speed faster. In contrast to transformer-based methods, this method is able to reconstruct images in a compressed level, producing more reliable and detailed results than previous methods. The denoising model is a time-conditioned U-Net, augmented with the cross-attention mechanism to handle flexible conditioning information for image generation (e.g. class labels, semantic maps, blurred variants of an image).

Without an independent classifier f_ϕ, it is still possible to run conditional diffusion steps by incorporating the scores from a conditional and an unconditional diffusion model [Ho and Salimans, 2022]. The unconditional denoising diffusion model $p_\theta(x)$ parameterized through a score estimator $\epsilon_\theta(x_t, t)$, together with the conditional model they can be learned via a single neural network. Their experiments showed that classifier-free model can achieve a good balance between FID (distinguish between synthetic and generated images) and IS (quality and diversity).

2 Preliminary

Non-IID Data in Federated Learning: In federated learning, data distributions across clients are often not identical and independent distributed (iid). In experiment, I use lable-skew non-iid federated learning setting, which assumes each client's training examples are drawn with class labels following a dirichlet distribution [Li et al., 2022], $\beta > 0$ is the concentration parameter controlling the identicalness among users.

Distribution-based Label Imbalance: A way to simulate label imbalance is that each party is allocated a proportion of the samples of each label according to Dirichlet distribution. Specifically, sample $p_k \sim Dir_N(\beta)$ and allocate a $p_{k,j}$ proportion of the instances of class k to party j. Here $Dir(\cdot)$ denotes the Dirichlet distribution and β is a concentration parameter ($\beta \geq 0$). An advantage of this approach is that it can flexibly change the imbalance level by varying the concentration parameter β.

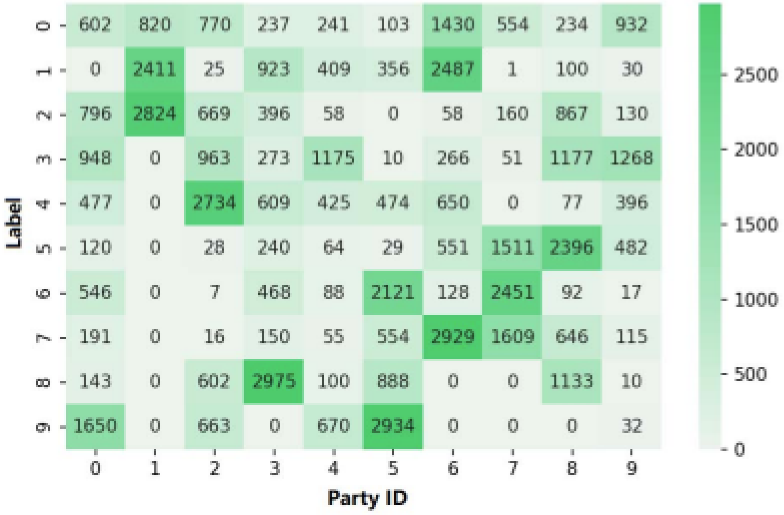

Figure 1: An example of distribution-based label imbalance partition on MNIST[LeCun et al., 1998] dataset with $\beta = 0.5$. The value in each rectangle is the number of data samples of a class belonging to a certain party.

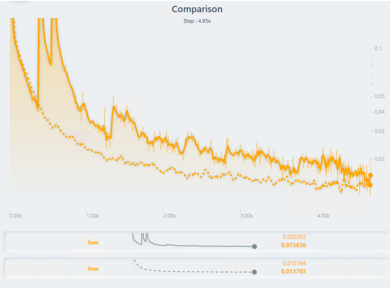

Figure 2: Learning Scheme Comparison (5 epochs)

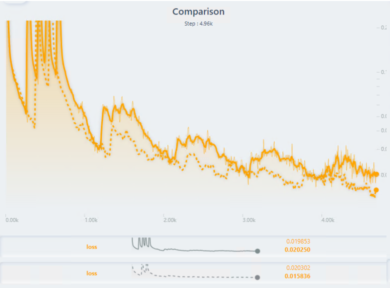

Figure 3: Clients Number Comparison (5 epochs)

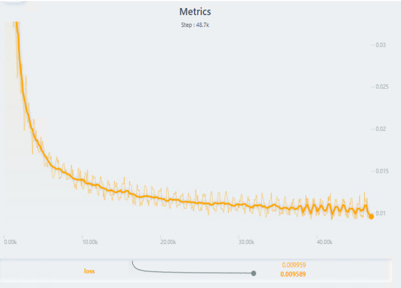

Figure 4: Federated Learning (50 epochs)

If β is set to a smaller value, then the partition is more unbalanced. An example of such a partitioning strategy [Li et al., 2022] is shown in Figure 4.

Federated Averaging (FedAvg): FedAvg [McMahan et al., 2017] is the most popular existing and easiest to implement federated learning strategy, where clients collaboratively send updates of locally trained models to a global server. Each client runs a local copy of the global model on its local data. The global model's weights are then updated with an average of local clients' updates and deployed back to the clients. This builds upon previous distributed learning work by not only supplying local models but also performing training locally on each device.

3 Experiment

Dataset and Setup: I conducted experiments on MNIST dataset [LeCun et al., 1998] and realized diffusion model based on labml [Varuna Jayasiri, 2020]. The number of clients is set to 3, 5 and each has a Dirichlet distribution dataset, and the rest of the experiment settings are in the table 1. Code is available at https://github.com/kyrie-23/federated_diffusion.

Overviews: The following paragraphs present a comprehensive investigation on the properties of the proposed federated approach, including: (1) convergence rate under different learning schemes; (2) generated figures at different level of epochs.

Convergence Rate: I analyze the training loss curve of FedAvg compared to Centralized as

Table 1: Experiment settings

batch_size	64
channel_muiltipliers	[1,2,24]
dataset	MNIST
image_channels_input	1
image_channels_output	3
image_size	32
is_attention	[0,0,0,1]
learning_rate	0.00002
n_channels	64
n_samples	16
n_steps	1000
n_clients	3,5
local_iters	1
optimizer	Adam

shown in Fig. 2. The loss of FedAvg does not drop off as fast as centralized learning at first, and it is not as smooth as model aggregation introduces different features from other clients. It drops faster after a few epochs, eventually reaching the same level as centralized learning. Comparing the number of different clients, as shown in Fig. 3, the more clients that participate, the less smooth it starts to drop. Also, more clients may need more epochs to smooth out the drop in loss because they need time to share the global model and they converge to the centralized model due to the FedAvg scheme. As shown in Fig. 4, the loss drops smoothly

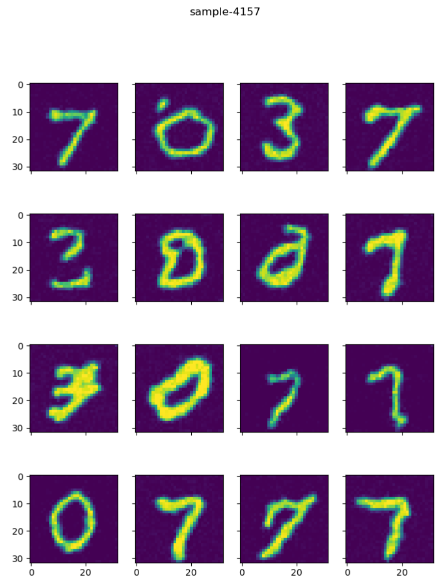

Figure 5: 5 epochs

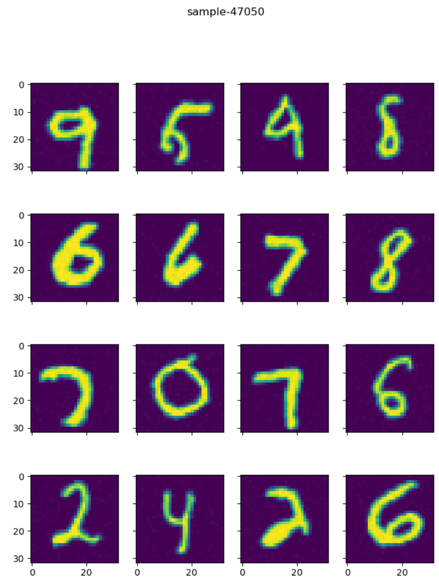

Figure 6: 50 epochs

Sample Quality: Fig. 5 and Fig. 6 are colored samples of 5 and 50 epochs, and I also generated videos with denoising animation. Unfortunately I haven't implemented quality metric evaluation this time (e.g. Inception scores, FID scores, negative log likelihoods).

Model Architecture: U-Net model for Denoising Diffusion Probabilistic Models (DDPM) to predict noise $\epsilon_\theta(x_t, t)$. There are pass-through connection at each resolution (as shown in Fig. 7. This implementation contains a bunch of modifications to original U-Net (residual blocks, multi-head attention) and also adds time-step embeddings t [Varuna Jayasiri, 2020].

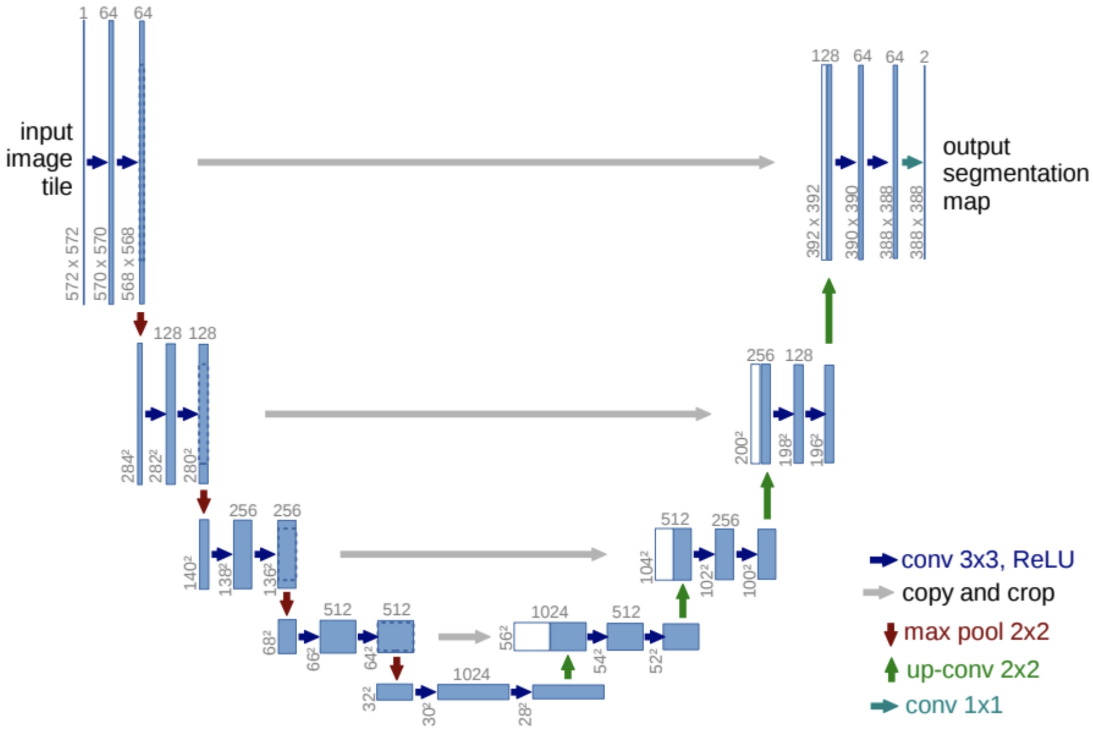

Figure 7: U-net architecture (example for 32x32 pixels). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

References

[Ho et al., 2020] Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851.

[Ho and Salimans, 2022] Ho, J. and Salimans, T. (2022). Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*.

[LeCun et al., 1998] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

[Li et al., 2022] Li, Q., Diao, Y., Chen, Q., and He, B. (2022). Federated learning on non-iid data silos: An experimental study. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 965–978. IEEE.

[McMahan et al., 2017] McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR.

[Nichol and Dhariwal, 2021] Nichol, A. Q. and Dhariwal, P. (2021). Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR.

[Rombach et al., 2022] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695.

[Song et al., 2020] Song, J., Meng, C., and Ermon, S. (2020). Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

[Varuna Jayasiri, 2020] Varuna Jayasiri, N. W. (2020). labml.ai annotated paper implementations.

[Weng, 2021] Weng, L. (2021). What are diffusion models? *lilianweng.github.io*.